

Japanese Patent Laid-open No. HEI 11-104357 A

Publication date : April 20, 1999

Applicant : K.K. Sega Enterprises

Title : GAME DEVICE AND RECORDING MEDIUM HAVING GAME

5 PROGRAM RECORDED THEREIN

(57) [Abstract]

[Object] To provide a game device that can be enjoyed by players of a wide range of age groups through simple operations of switches.

10 [Means] In a game device that has a game program and operation buttons, and displays game images on a display screen, this invention is characterized by displaying subject blocks 20 of an arbitrary shape on the display screen, changing a shape of moving block objects 22 that move in a predetermined direction in the display screen when the operation buttons 14 and 15 are  
15 operated, determining whether the changed shape of the moving block objects matches the shape of the subject blocks, and if so, determining that the game is cleared, and further determining whether the moving blocks have traveled to a predetermined position before they are processed into the shape of the subject blocks, and if so, determining that the game is over.

20

[0012] The outline of the game of this embodiment is, as shown in the liquid crystal display unit 12 of Fig.1, for instance,  $5 \times 5$  matrix-shaped blocks drop from top to bottom of the display screen and are processed into the shape of the subject blocks 20. The processing of these falling blocks is achieved by  
25 having missiles 26 fired by the player 24 that can move right and left to the

**THIS PAGE BLANK (USPTO)**

lower part of the screen hit the blocks, and clearing them away. To this end, the player manipulates right/left and up/down operation keys 16, the key for firing the missiles 26, and the key for rotating the falling  $5 \times 5$  drop blocks.

The latter key is achieved by the operation switch 14.

5 [0013] Thus, the player recognizes the shape of the subject blocks 20 shown on the upper left part of the screen, and clears the falling blocks one by one by using the missiles 26 while rotating the falling blocks, by the time the falling blocks 22 reach the lower part of the screen. If the player cannot process the shape of the blocks into that of the subject blocks before the falling blocks drop,  
10 he/she fails (loses) the game and this game of the falling blocks is over.

Alternatively, when the player inadvertently clears blocks that should not be cleared, the falling block game is also over.

[0014] Since many types of subject blocks 20 are prepared and selected at random, a player should process the falling blocks into various shapes, which  
15 can thus give the player more intellectual stimulation. In addition, changing shapes of the falling blocks or varying the number of the blocks, etc., can have the player use his/her brain and thus excite his/her interest. Selection of the subject blocks is not necessarily at random, and the blocks may also be selected in a predetermined sequence.

20 [0015] Fig. 2 illustrates an example of an internal configuration of the game device of this embodiment. A CPU 30 that executes the game program, a main memory 32 that stores parameters necessary for execution of the game program, and a cartridge connector 34 that mounts a game cartridge 36 having a storage medium that contains the game program are connected by way of a  
25 bus BUS. To the bus BUS are also connected a video processor 38 that

**THIS PAGE BLANK (USPTO)**

generates and outputs display data, and a periphery control unit 42. To the video processor 38 is connected a video memory 40 that stores display data of at least one frame. In addition, to the periphery control unit 42 are connected a control pad 48 that constitutes operation keys, a communications connector 50, a liquid crystal driver 44, etc. The liquid crystal driver 44 supplies a drive signal to a liquid crystal display unit 12 according to the display data to be output from the video processor 38.

[0016] The game program and game data recorded in the game cartridge 36 are transferred to a main memory 32 and executed by the CPU30. The CPU 30 inputs an operation signal from the control pad 48, executes the game program, makes necessary changes to display data, and displays game screens on the liquid crystal display unit. In addition, parameter data needed during execution of the game is recorded in the main memory 32.

[0017] Before description of specific content of the game program, details of the block processing game of this embodiment is described. Fig. 3 illustrates examples of the screens as the block processing game progresses.

[0018] Fig. 3 (1) is a view of the first screen of the game. A shape of subject blocks 20 is displayed on the upper left part of the screen, and falling blocks 22 sized, for instance,  $5 \times 5$  are falling from the center top of the screen. A player 24, which can fire missiles, appears in the center bottom of the screen. As shown in Fig. 3 (2), a player manipulates the right/left and up/down keys 16 to move the player 24 from side to side, fires missiles 26 from the player by manipulating A key 14, and clears away the falling blocks 22 one by one. In addition, as shown in Fig. 3 (3), the player manipulates B key 14 to turn the falling block 22 clockwise by 90 degrees and steer the blocks to be cleared

**THIS PAGE BLANK (USPTO)**

toward a direction of the player 24.

[0019] As described above, while manipulating the 3 operation keys, the player has the missiles 26 hit the falling blocks 22 to clear them away, thereby processing the falling blocks 22 into a shape of the subject blocks 20. Then, as shown in Fig. 3 (4), when the falling blocks are processed into the shape identical to that of the subject blocks before the falling blocks 22 reach the bottom of the screen, the subject is cleared, new subject blocks appear, and unprocessed  $5 \times 5$  falling blocks begin to drop from the upper part of the screen.

10 [0020] In this embodiment, a game becomes over when the falling blocks 22 drop before they are processed into the shape of the subject blocks 20, and also when the missiles 26 clear the blocks that should not be cleared away. In other words, as shown in Fig. 3 (5), the game will be lost and over, when the blocks in the center of the falling blocks 22 are cleared, which means that processing into the shape of the subject blocks 20 has become impossible. In addition, as shown in Fig. 3 (6), if the falling blocks reach the bottom of the screen before they are processed into the shape of the subject blocks 20, the game will be lost and over.

[0021] When the game is lost and over, as shown in Fig. 7 (3), the falling blocks are disassembled and seen to scatter in  $360^\circ$ . At this point of time, the game can be over. Otherwise, the game can be over when the scattering blocks hit the player 24. The latter method can vary the probability of game over depending on a how far the blocks have been processed. In addition, setting the number of available players to a predetermined number allows the game to be over when the predetermined number of players is lost. Needless

**THIS PAGE BLANK (USPTO)**



to say, some contrivance can be added to the game, such as increased number of players by clearing a predetermined number of subjects or clearing a bonus subject, etc.

**THIS PAGE BLANK (USPTO)**

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-104357

(43) 公開日 平成11年(1999) 4月20日

(51) Int.Cl.<sup>9</sup>

A 6 3 F 9/22

識別記号

F I

A 6 3 F 9/22

S

H

審査請求 未請求 請求項の数9 O L (全 14 頁)

(21) 出願番号 特願平9-272891  
(22) 出願日 平成9年(1997)10月6日

(71) 出願人 000132471  
株式会社セガ・エンタープライゼス  
東京都大田区羽田1丁目2番12号  
(72) 発明者 島村 誠人  
東京都大田区羽田1丁目2番12号 株式会  
社セガ・エンタープライゼス内  
(72) 発明者 小寺 攻  
東京都大田区羽田1丁目2番12号 株式会  
社セガ・エンタープライゼス内  
(74) 代理人 弁理士 土井 健二 (外1名)

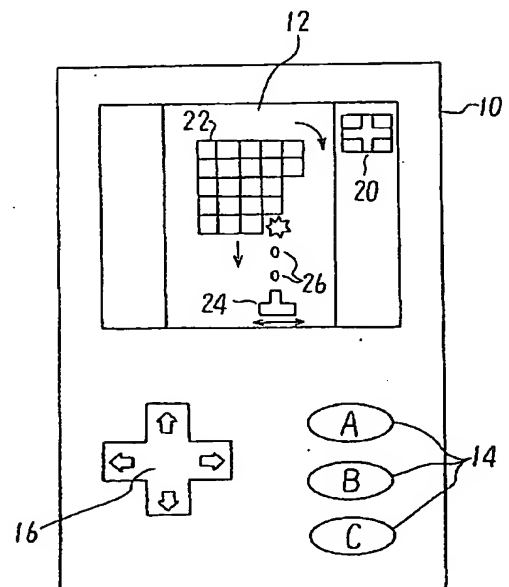
(54) 【発明の名称】 ゲーム装置及びゲームプログラムを記録した記録媒体

(57) 【要約】

【課題】 簡単なスイッチ操作で広範な年齢層にも楽しむことができるゲーム装置を提供する。

【解決手段】 本発明は、ゲームプログラムと操作ボタンを有し、表示画面にゲーム映像を表示させるゲーム装置において、表示画面に任意の形状の問題ブロック20を表示し、操作ボタン14、16を操作することにより表示画面内の所定方向に移動する移動ブロック体22の形状を変更させ、変更された移動ブロック体の形状が前記問題ブロックの形状と一致するか否かを判定し、一致した時にゲームクリアと判定し、更に、移動ブロック体が問題ブロックの形状に加工される前に所定の位置まで移動したか否かを判定し、所定位置まで移動した時にゲームオーバーと判定することを特徴とする。

ゲーム装置の外観図



## 【特許請求の範囲】

【請求項 1】 ゲームプログラムと操作ボタンを有し、表示画面に対してゲーム画像を表示させるゲーム装置において、  
 任意の形状の問題ブロックを生成する問題ブロック制御手段と、  
 前記表示画面内の所定方向に移動ブロック体を移動させるブロック制御手段と、  
 前記操作ボタンからの入力信号に応じて、前記表示画面内のプレーヤ体を移動させ、前記プレーヤ体からミサイル体を発射させるプレーヤ体制御手段と、  
 前記発射されたミサイル体を移動させ、前記移動ブロック体に衝突したか否かを検出し、衝突した時に移動ブロック体の形状を変更するミサイル体制御手段と、  
 前記移動ブロック体が前記問題ブロックの形状と一致するか否かを検出し、一致する時にゲームクリアとするブロック照合手段とを有することを特徴とするゲーム装置。

【請求項 2】 請求項 1 において、  
 前記問題ブロック制御手段は、生成された問題ブロックから複数方向の照合用の問題ブロックデータを生成し、前記ブロック照合手段は、前記移動ブロック体と前記複数の照合用の問題ブロックとを比較することを特徴とするゲーム装置。

【請求項 3】 請求項 1 において、  
 前記ブロック照合手段は、前記移動ブロック体が前記問題ブロックの形状に加工するに修復不可能な形状になったか否かを更に検出し、  
 更に、前記ブロック制御手段は、前記移動ブロック体が修復不可能な形状になったことが検出された時に前記移動ブロック体を飛散させることを特徴とするゲーム装置。

【請求項 4】 請求項 1 において、  
 前記ブロック制御手段は、前記移動ブロック体が所定位置に移動したことを検出し、移動したことが検出された時に前記移動ブロック体を飛散させることを特徴とするゲーム装置。

【請求項 5】 請求項 3 または 4 において、  
 前記プレーヤ体制御手段は、前記飛散された移動ブロック体のブロックが、前記プレーヤ体に衝突したか否かを検出し、衝突した時にゲームオーバーとすることを特徴とするゲーム装置。

【請求項 6】 請求項 1 ～ 5 のいずれかにおいて、  
 更に、前記操作ボタンからの入力信号に応じて、前記表示画面内の移動ブロック体を回転させる移動ブロック体回転手段を有することを特徴とするゲーム装置。

【請求項 7】 ゲームプログラムと操作ボタンを有し、表示画面に対してゲーム画像を表示させるゲーム装置において、  
 前記表示画面に任意の形状の問題ブロックを表示し、

前記操作ボタンを操作することにより前記表示画面内の所定方向に移動する移動ブロック体の形状を変更させ、前記変更された移動ブロック体の形状が前記問題ブロックの形状と一致するか否かを判定し、一致した時にゲームクリアと判定し、

更に、前記移動ブロック体が前記問題ブロックの形状に加工される前に所定の位置まで移動したか否かを判定し、移動した時にゲームオーバーと判定することを特徴とするゲーム装置。

10 【請求項 8】 遊戯者からの操作入力に応じてコンピュータに表示画面を変更させるゲームプログラムを記録した記録媒体において、

前記表示画面に任意の形状の問題ブロックを表示させる手順と、

前記操作ボタンを操作することにより前記表示画面内の所定方向に移動する移動ブロック体の形状を変更させる手順と、

20 前記変更された移動ブロック体の形状が前記問題ブロックの形状と一致するか否かを判定し、一致した時にゲームクリアと判定する手順と、

更に、前記移動ブロック体が前記問題ブロックの形状に加工される前に所定の位置まで移動したか否かを判定し、移動した時にゲームオーバーと判定する手順とをコンピュータに実行させることを特徴とするゲームプログラムを記録した記録媒体。

30 【請求項 9】 請求項 8 において、  
 更に、前記生成された問題ブロックから複数方向の照合用の問題ブロックデータを生成させる手順を有し、  
 前記移動ブロック体が前記問題ブロックの形状と一致するか否かを検出を、前記移動ブロック体と前記複数の照合用の問題ブロックとを比較することにより実行させることを特徴とするゲームプログラムを記録した記録媒体。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 本発明は、ゲーム装置に関し、単純で広範な年齢層が楽しむことができるゲーム装置及びそれを実行するゲームプログラムを記録した記録媒体に関する。

## 40 【0002】

【従来の技術】 表示画面上で次々に落下してくる所定の幾何学形状を有する落下物を、所定の角度に回転し所定の位置に移動して表示画面内で一定の固まりを構成するゲームや、次々に落下してくる所定の形状を有する落下物を、所定の位置に移動して表示画面内で隣接した位置に配置させることでその隣接体を消滅されるゲーム等、単純で広範な年齢層が楽しめるゲーム装置が開発されている。

## 50 【0003】

【発明が解決しようとする課題】 しかしながら、これら

のゲーム装置では、落下してくる落下物の形状に何らかの変化を与えることは行わずに、単に落下物の角度や左右の位置を変化させるものである。従って、遊戯者はある程度の時間かかるゲームを楽しむと、飽きが来てしまう。そこで、単純ではあるがより知的で楽しみが継続するゲーム装置が望まれる。

【0004】そこで、本発明の目的は、簡単なスイッチ操作で広範な年齢層にも楽しめることができるゲーム装置及びそのゲームを実行するプログラムを記録した記録媒体を提供することにある。

【0005】更に、本発明の目的は、落下物の形状に何らかの変化を与えるという新しい発想を利用して、簡単なスイッチ操作で広範な年齢層にも楽しめることができるゲーム装置及びそのゲームを実行するプログラムを記録した記録媒体を提供することにある。

【0006】

【課題を解決するための手段】上記の目的を達成する為に、本発明は、ゲームプログラムと操作ボタンを有し、表示画面に対してゲーム画像を表示させるゲーム装置において、前記表示画面に任意の形状の問題ブロックを表示し、前記操作ボタンを操作することにより前記表示画面内の所定方向に移動する移動ブロック体の形状を変更させ、前記変更された移動ブロック体の形状が前記問題ブロックの形状と一致するか否かを判定し、一致した時にゲームクリアと判定し、更に、前記移動ブロック体が前記問題ブロックの形状に加工される前に所定の位置まで移動したか否かを判定し、移動した時にゲームオーバーと判定することを特徴とする。

【0007】画面内の例えば落下方向に移動する移動ブロック体に対して、遊戯者は、問題ブロックの形状になる様に操作ボタンを操作してその形状を変更させ、移動ブロック体が画面の所定の位置に移動する前にその形状を問題ブロックの形状に加工することができるか否かのスリルを味わうことができるので、単純ではあるが広範な年齢層にも楽しめることができるゲーム装置を提供することができる。

【0008】更に、本発明のゲーム装置は、移動ブロック体と問題ブロックとの照合を効率的に行う為に、問題ブロックが生成された時に、その問題ブロックの複数方向の形状のデータを予め生成し、移動ブロック体が移動中はその複数方向の形状のデータと照合することで、短い演算工程で移動ブロック体が問題ブロックと一致したか否かの照合を行うことができる。

【0009】上記の目的を達成する為に、本発明は、遊戯者からの操作入力に応じてコンピュータに表示画面を変更させるゲームプログラムを記録した記録媒体において、前記表示画面に任意の形状の問題ブロックを表示させる手順と、前記操作ボタンを操作することにより前記表示画面内の所定方向に移動する移動ブロック体の形状を変更させる手順と、前記変更された移動ブロック体の

形状が前記問題ブロックの形状と一致するか否かを判定し、一致した時にゲームクリアと判定する手順と、更に、前記移動ブロック体が前記問題ブロックの形状に加工される前に所定の位置まで移動したか否かを判定し、移動した時にゲームオーバーと判定する手順とをコンピュータに実行させることを特徴とする。

【0010】

【発明の実施の形態】以下、本発明の実施の形態について図面に従って説明する。しかしながら、本発明の技術的範囲がその実施の形態に限定されるものではない。

【0011】図1は、本実施の形態例のゲーム装置の例の外観図である。手のひらに乗る程度の大きさの筐体10の表面に、液晶表示部12、操作スイッチ14、左右上下操作キー16等を備える。また、図示されないが、筐体の裏面側には、ゲームプログラムを記録した媒体を有するゲームカートリッジが装着される。

【0012】本実施の形態例のゲームの概略は、図1の液晶表示部12に示される通り、表示画面の左上に表示される問題ブロック20の形状になるように、表示画面上部から下部に落下してくる例えば5×5のマトリクス状のブロックを加工することにある。この落下ブロックの加工は、画面下側に左右に移動可能なプレーヤ24から発せられるミサイル26をブロックに衝突させて、ブロックを消滅させることで行われる。そのために、プレーヤ24を左右に移動させる左右上下操作キー16と、ミサイル26を発するキー及び落下する5×5の落下ブロックを回転させるキーとが、遊戯者により操作される。上記後者のキーは、操作スイッチ14により実現される。

【0013】従って、遊戯者は、画面の左上に表示される問題ブロック20の形状を認識し、落下ブロック22が画面の下側に落下するまでに、落下ブロックを回転させながらミサイル26を利用して一つ一つのブロックを消滅させる。落下ブロックが落下するまでに問題ブロックの形状に加工できなかった場合は、失敗（ミス）となり、この落下ブロックのゲームは終了する。或いは、消滅させてはならないブロックを誤って消滅させた場合も、この落下ブロックのゲームは終了する。

【0014】問題ブロック20を多数種類準備し、ランダムに選択することで、遊戯者は、落下してくる落下ブロックを様々な形状に加工しなければならず、より知的な興奮を味わうことができる。また、落下ブロックの形状、ブロック数等を種々変更することにより、遊戯者に様々な工夫をさせ、遊戯者の興味をそそることができる。尚、上記の問題ブロックの選択は、ランダムである必要はなく、例えば予め決められた順番で選択されてもよい。

【0015】図2は、本実施の形態例のゲーム装置の内部構成の例を示す図である。ゲームプログラムを実行するCPU30、ゲームプログラムの実行に際して必要な

パラメータを保存するメインメモリ32、ゲームプログラムが格納された記録媒体を有するゲームカートリッジ36を装着する為のカートリッジコネクタ34が、バスBUSを介して接続される。また、バスBUSには、表示データを生成して表示データを出力するビデオプロセッサ38、周辺機器制御部42も接続される。ビデオプロセッサ38には、少なくとも1フレーム分の表示データを記録するビデオメモリ40が接続される。更に、周辺機器制御部42には、操作キーであるコントロールパッド48、通信コネクタ50、液晶ドライバ44等が接続される。液晶ドライバ44は、ビデオプロセッサ38から出力される表示データに従って、駆動信号を液晶表示部12に供給する。

【0016】ゲームカートリッジ36に記録されたゲームプログラム及びゲームデータが、メインメモリ32に転送され、CPU30により実行される。CPU30は、コントロールパッド48からの操作信号を入力し、ゲームプログラムを実行し、必要な表示データの変更を行い、液晶表示部にゲーム画面を表示する。また、ゲーム実行中に必要なパラメータデータは、メインメモリ32に記録される。

【0017】ゲームプログラムの具体的な内容を説明する前に、本実施の形態例のブロック加工ゲームの詳細を説明する。図3は、ブロック加工ゲームの進行に伴う画面の例を示す図である。

【0018】図3(1)は、ゲームの最初の画面を示す。画面の左上に問題ブロック20の形状が表示され、画面中央の上部から例えば5×5の落下ブロック22が落下してくる。画面中央下部には、ミサイルを発射することができるプレーヤ24が表示される。図3(2)に示される通り、落下ブロック22に対して、遊戯者は、左右上下操作キー16を操作してプレーヤ24を左右に移動させ、Aキー14を操作してプレーヤ24からミサイル26を発射させて、落下ブロック22を1個ずつ消滅させる。更に、図3(3)に示される通り、遊戯者はBキー14を操作して落下ブロック22を90°単位で時計方向に回転させ、消滅させるべきブロックをプレーヤ24の方向に向ける。

【0019】上記の通り、遊戯者は3つの操作キーを操作しながら、落下ブロック22にミサイル26を衝突させてブロックを消滅させ、落下ブロック22を問題ブロック20の形状に加工する。そして、図3(4)に示される通り、落下ブロック22が画面の下まで落下する前に、問題ブロックの形状と同じ形状に加工されると、その問題はクリアとなり、新たな問題ブロックが表示され未加工の5×5の落下ブロックが画面上部から落下し始める。

【0020】本実施の形態例では、ゲームオーバーは、落下ブロック22が問題ブロック20の形状に加工される前に落下してしまう場合と、ミサイル26により落下ブ

ロック22の消滅すべきでないブロックが消滅した場合とに発生する。即ち、図3(5)に示される通り、落下ブロック22の真ん中のブロックが消滅されてしまい、問題ブロック20の形状への加工が不可能になった時点で、ミスとなり、ゲームオーバーになる。また、図3

(6)に示される通り、落下ブロック22が問題ブロック20の形状に加工される前に、画面下部に落下してしまうと、ミスとなりゲームオーバーになる。

【0021】ミスによりゲームオーバーになると、図3(7)に示される様に、落下ブロックが分解され、360°の方向に飛び散るように表示される。この時点で、ゲームオーバーとすることができる。或いは、飛び散るブロックがプレーヤ24に衝突した時点でゲームオーバーにすることもできる。後者の方法をとることで、ブロックの加工の程度に応じてゲームオーバーになる確率を変えることができる。また、使用できるプレーヤの数を所定の数に設定することで、その所定数のプレーヤ分だけミスをするゲームオーバーにすることもできる。勿論、所定数の問題をクリアしたり、ボーナス問題をクリアすることで、プレーヤの数を増加させる等の工夫を加えることもできる。

【0022】図4は、問題ブロックの他の例を示す図である。問題1は対向する頂点のブロックが消滅している例、問題2は4つの頂点部のブロックが消滅している例、問題3は漢数字「七」に類似した非対称形の例、問題4は階段状の形状の例である。ここに示した以外の問題ブロックも、種々考えられる。

【0023】本実施の形態例のゲーム装置において、特徴的な点は、加工される落下ブロックと問題ブロックとが一致しているか否かを逐次検出することにある。その為に、本実施の形態例では、ゲームプログラムの実行中に、問題ブロックのマトリクスデータと、加工された落下ブロックのマトリクスデータとを照合する工程を有する。その際に、加工された落下ブロックの向きは、必ずしも問題ブロックと同じである必要がなく、図4に示された問題3、4の様に問題ブロックが左右、上下に非対称形である場合は、問題ブロックの4方向のマトリクスデータと落下ブロックのマトリクスデータとが照合され、いずれかで一致する場合は、落下ブロックが問題ブロックの形状と一致したとの判断がなされる。

【0024】更に、特徴的な点は、加工される落下ブロックが消滅すべきでない修復不可能なブロックを消滅させてしまったか否かを、問題ブロックのマトリクスデータと照合することで検出することにある。その為に、上記と同様に、問題ブロックに対して4方向のマトリクスデータを生成し、それらのマトリクスデータと落下ブロックのマトリクスデータとを照合し、問題ブロックのマトリクスデータに存在し、落下ブロックのマトリクスデータに存在しないブロックが、全ての照合の結果検出された時に、消滅すべきでないブロックが消滅されたとの

判断がなされる。

【0025】更に、特徴的なことは、落下ブロックが問題ブロックと一致する前に画面の下部まで落下してしまったか否かを検出する点にある。この検出は、落下ブロックの位置データを常に更新し、その位置データが所定の位置以下になったか否かを検出することで行われる。

【0026】図5は、問題ブロックデータと照合用に生成される問題ブロックデータとを示す図である。図5

(1)の問題ブロックデータは、図4の問題4の階段状の問題ブロックのデータ例を示す。ブロックが存在する領域を「1」に、存在しない領域を「0」にする構成である。かかる問題ブロックに対して、その方向を0°、90°、180°、270°回転させた場合の照合用の問題ブロックデータが、図5(2)、(3)、(4)、(5)に示される。それぞれ、データ「1」「0」の位置が回転した位置にあることが理解される。

【0027】問題ブロックデータから、それぞれ方向を回転させた照合用の問題ブロックデータを生成する為には、一定の演算処理が必要であるので、問題ブロックが選択された時点で、上記の如き照合用の問題ブロックデータが生成され、メインメモリに保存される。問題ブロックデータは、ゲームプログラムカートリッジからメインメモリ内に転送されたプログラムデータ内にあり、かかる選択された問題ブロックデータから照合用の4種類の問題ブロックデータが生成される。

【0028】図6は、メインメモリ内のデータ構成例を示す図である。ここに示された以外に、ゲームプログラムのカートリッジから転送されたプログラムも保存されているが、図6では省略されている。この例では、上記した照合用の問題ブロックデータ60が4種類記録されている。更に、後に説明するサブルーチンプログラム管理用のフラグ62、落下ブロックデータ64、プレーヤの数データ、位置データ66、ミサイルの位置データ68が記録される。落下ブロックデータ64には、その角度データ(0°90°180°270°)、落下の位置データ、5×5のビットデータが含まれる。

【0029】図7はゲームプログラム全体のフローチャート図である。また、図8はイベントサブルーチンのフローチャート図である。更に、図9、図10はそれぞれプレーヤサブルーチンとミサイルサブルーチンのフローチャート図である。図7のゲームプログラム全体のフローチャートに従って説明すると、ゲームの開始時に、初期値が設定され、画像データがメインメモリにロードされる(S1)。初期値としては、例えば、複数の問題ブロックデータ、プレーヤの数等である。そして、イベントサブルーチン(S2)、プレーヤサブルーチン(S3)、ミサイルサブルーチン(S4)、表示サブルーチン及び入力サブルーチン(S5)が繰り返し実行される。そして、イベントサブルーチンにてゲーム終了となると、全てのデータがクリアされ(S6)、ゲームが終

了する。

【0030】イベントサブルーチンには、図8に示される通り、タイムアウトルーチン、ニュー問題ルーチン、ブロックチェックルーチン、ブロック爆発ルーチン、ミスルーチンが含まれ、タイムアウトルーチンは常に実行されるが、他のルーチンはメインメモリ内のそれぞれの実行フラグが実行状態の時に実行される。但し、ニュー問題ルーチンは、ゲーム開始時に実行される。それぞれのルーチンについては後述する。

10 【0031】プレーヤサブルーチンは、プレーヤ24の制御を行うサブルーチンであり、このサブルーチンはフラグの状態にかかわらず毎回実行される。そして、ミサイルサブルーチンは、ミサイルの移動と落下ブロックとの衝突を制御するサブルーチンであり、ミサイルが発射された時にミサイルフラグが「1」になると実行され、ミサイルが画面の外にでるまで実行される。

20 【0032】図7のゲームプログラム全体のフローチャートに示される通り、各サブルーチンが繰り返し実行され、サブルーチンの実行により生成されたゲーム進行中に変更されるメインメモリ内のデータに従って、表示サブルーチンで画面12への表示データの生成を行う。また、コントロールパッド48からの入力信号のチェックを行う。

30 【0033】図8に示されたイベントサブルーチンについて説明する。タイムアウトルーチンは、ステップS11～S14で構成され、落下ブロックの位置を毎回下げて(S11)、落下ブロックが画面の最下位置に達したか否かのチェックを行う(S12)。落下ブロックの位置を毎回下げる為に、メインメモリ内の落下ブロックデータの位置データを変更する。また、落下ブロックが最下位置に達したか否かは、その位置データと所定の閾値データとの比較を行うことで検出される。最下位置に達すると、ブロック爆発フラグを「1」にしてタイムアウトルーチンを終了する(S13)。また、最下位置に達していなければ、そのままタイムアウトルーチンを終了する(S14)。

40 【0034】タイムアウトルーチンは、イベントサブルーチンが実行される時に毎回実行されるが、それ以外のブロック爆発ルーチン、ブロックチェックルーチン、ニュー問題ルーチン及びミスルーチンは、それぞれ対応するフラグが実行状態を示す「1」の状態の時に実行される。その制御は、例えばステップS10にて行われる。即ち、タイムアウトルーチンとフラグが実行状態にある他のルーチンが、イベントサブルーチンで実行される。

50 【0035】ゲームの開始時は、強制的にニュー問題ルーチンが実行される。このルーチンでは、例えば乱数表に従って問題ブロックが選択される(S16)。そして、その選択された問題ブロックデータから、照合用の正解となるブロックデータを4方向に分作成し、メインメモリ32に保存する(S17)。そして、ニュー問題フ

ラグを「0」にしてから、ニュー問題ルーチンを終了する(S18)。ニュー問題ルーチンは、落下ブロックが加工されて照合用のブロックデータと一致してクリアになった後でも実行される。その場合は、後述する通りクリアになった時点でニュー問題ルーチンのフラグが実行状態「1」されることで、ステップS10にて選択実行される。

【0036】ブロックチェックルーチンは、プレーヤ24から発射されたミサイル26が落下ブロックに衝突してブロックが消滅した時に実行されるルーチンである。従って、ミサイルが衝突した後以外では、このルーチンは実行されない。ブロックチェックルーチンは、まず落下ブロックのビットデータと、4方向について生成された照合用の問題ブロックデータ60とがそれぞれ照合される(S20)。この照合において、メインメモリ32内のそれぞれのデータを利用して行われる。いずれの照合においても、落下ブロックのビットデータでは存在しないビットが、4方向の照合用の問題ブロックデータでは存在するビットとして判断された場合は、消去すべきでないブロックを消去したと判断される(S21)。その場合は、ブロック爆発フラグを「1」の状態にして、ルーチンを終了する(S22)。

【0037】更に、落下ブロックのビットデータと4方向の照合用のブロックデータとのいずれかの照合で、全てのビットが一致したことが検出されると(S23)、ニュー問題フラグが「1」の状態にされて、ルーチンが終了する(S24)。不一致の場合は、ブロックチェックフラグを「0」にしてルーチンが終了する(S25)。

【0038】ブロック爆発ルーチンは、上記のステップS13及びS22にてブロック爆発フラグが「1」になっている場合に実行される。即ち、落下ブロックが画面の最下位置まで落ちた時と、消去すべきでないブロックが消去された場合である。ブロック爆発ルーチンでは、落下ブロックを構成するそれぞれのブロックを、いずれかの方向にランダムに画面の外に向かって移動する(S27)。即ち、メインメモリ32内の落下ブロックデータ64における各ブロックの位置データを変更する。そして、全てのブロックの位置データが画面の外側に移動すると(S28)、ブロック爆発フラグを「0」にし、ニュー問題フラグを「1」にしてからこのルーチンを終了する(S29)。いずれかのブロックの位置が画面内にある時は、そのままルーチンを終了する(S30)。

【0039】更に、イベントサブルーチン内のミスルーチンは、爆発したブロックがプレーヤ24に衝突した時に実行されるルーチンであり、プレーヤの数がなくなってゲームオーバー状態か否かが判断され(S32)、ゲームオーバーであればミスフラグを「0」に戻してゲームタイトル画面を表示する(S33)。また、ゲームオーバーでなければ、ミスフラグを「0」にしてこのルーチン

を終了する(S34)。ゲーム上のミスは、上記の場合に限定されず、例えば落下ブロックが爆発した段階でミスにしてもよい。

【0040】上記した通り、イベントサブルーチンでは、新たな問題ブロックの選択が必要になった時、ミサイルが落下ブロックに衝突した時、ブロックが爆発する時、ゲームのミスが生じた時等のイベントを処理する。それ以外に、落下ブロックの処理も行う。

【0041】図9に示されたプレーヤサブルーチンでは、コントロールパネルの左右操作キー1,6からの入力データに従って、プレーヤの位置を移動させる(S41)。具体的には、メインメモリ32内のプレーヤの位置データを変更する。そして、プレーヤが爆発したブロックのいずれかに衝突したか否かの判定を行う(S42)。具体的には、メインメモリ内のプレーヤの位置データと各ブロックの位置データとが一致若しくは近接しているか否かの判定を行う。衝突している場合は、ミスとなり、ミスフラグを「1」にしてサブルーチンを終了する(S43)。

【0042】プレーヤサブルーチンでは、更に、入力データからミサイルが発射されたか否かの判定を行う(S44)。発射させる入力データがある場合は、ミサイルフラグを「1」にしてこのサブルーチンを終了する(S45)。発射されていなければ、そのままサブルーチンを終了する(S46)。プレーヤサブルーチンは、フラグ状態にかかわらず毎回実行される。

【0043】図10に示されたミサイルサブルーチンは、ミサイルフラグが「1」の時に実行される。先ず、ミサイルの位置データを所定距離移動させる(S51)。即ち、メインメモリ内のミサイル位置データを変更する。その後、ミサイルが落下ブロックに衝突したか否かの判定を行う(S52)。具体的には、メインメモリ内のミサイルの位置データと、落下ブロックのうち存在しているブロックの位置データとが比較される。一致していれば、ミサイルがブロックに衝突したことになり、そのブロックのビットデータを「0」に書き換え、ブロックチェックフラグを「1」にしてこのサブルーチンを終了する(S53)。

【0044】また、ミサイルが画面の外に移動したか否かの判定を行い(S54)、画面外であればそのミサイルのデータを消去してこのサブルーチンを終了する(S55)。また、画面内であればそのままサブルーチンを終了する(S56)。

【0045】以上が、ゲームプログラムがコンピュータで構成されるゲーム装置に実行させる処理である。ゲームプログラムがアセンブラなどのオブジェクトコードで記述される場合は、そのプログラムコードをCPU30が直接解読して上記の処理を実行する。また、ゲームプログラムがソースコードで記述される場合は、一旦オブジェクトコードにコンパイルされてから、上記の処理が



行われる。また、コンピュータからなるゲーム装置がオペレーションプログラムを有する場合は、ゲームプログラムの命令コードに従って、そのオペレーションプログラムと共同して、上記の処理が行われる。或いは、ゲーム装置の通信コネクタ50に接続された通信回線を介してゲームプログラムがメインメモリ内に直接保存される場合もある。

【0046】そこで、図3で示したゲーム画面の例に従って、上記したゲームプログラムの各処理がどのように実行されるかについて説明する。先ず、図3(1)では、イベントサブルーチンのニュー問題ルーチンにより、問題ブロック20が選択され、表示サブルーチン(S5)により、その問題ブロック20と共に、落下ブロック22とプレーヤ24とが表示される。

【0047】図3(2)に示される通り、遊戯者がプレーヤ24を左右に移動させるように上下左右操作ボタン16を操作し、ミサイル発射の為にAボタン14を操作すると、その操作が入力サブルーチン(S5)にて検出され、その操作に従ってプレーヤサブルーチンにてプレーヤ24の位置が変更され(S41)、ミサイルの発射が検出され(S44)、ミサイルサブルーチンにてミサイルの移動が制御される(S51)。そして、ミサイルサブルーチンにて、ミサイルが落下ブロックのいずれかのブロックに衝突したことが検出されると(S52)、イベントサブルーチン内のブロックチェックルーチンが実行される。

【0048】更に、図3(3)に示される通り、遊戯者が操作ボタンBを操作して落下ブロックを90°毎に回転させると、その操作入力が入力サブルーチンで検出され、それに従って落下ブロックのデータの角度データが変更される。そして、表示サブルーチンにて、その角度データに応じて落下ブロック22が表示される。

【0049】図3(4)に示される通り、ミサイルサブルーチンにて、発射したミサイルが落下ブロックに衝突したことが検出されると、イベントサブルーチン内のブロックチェックルーチンにて、落下ブロックのデータと照合用の問題ブロックのデータとが照合チェックされる(S29)。そして、落下ブロックのデータと照合用の問題ブロックのいずれかと一致したことが検出されると(S23)、クリアになり、次の問題ブロックがニュー問題ルーチンにて選択され(S16)、照合用の問題ブロックのデータが生成されると共に(S17)、表示サブルーチンにて新しい問題ブロックが表示画面の右上に表示される。

【0050】また、図3(5)に示される通り、ブロックチェックルーチンにて消去してはならないブロックの消去が検出されると(S21)、ブロック爆発ルーチンにて落下ブロック22が爆発する様にそれぞれのブロックの位置が変更される(S28)。そして、図3(7)に示される通り、爆発したブロックがプレーヤに衝突し

たことがプレーヤサブルーチンにて検出されると(S42)、ミスとなる。

【0051】或いは、図3(6)に示される通り、落下ブロック22が問題ブロックと一致する前に、表示画面の下部に達すると、イベントサブルーチン内のタイムアウトルーチンにて検出され(S12)、上記と同様にブロック爆発ルーチンにて落下ブロックの爆発が制御され、プレーヤにそのブロックが衝突したかの検出が行われる。

10 【0052】上記したゲームプログラムでは、ゲームの難易度に応じて問題ブロックの種類を変化させたり、落下時間を短くしたり、落下ブロックのブロック数を増加させたりすることができる。更に、移動ブロック体の選択は、予め決められた順番で表示されるようにしてもよい。その場合は、移動ブロック体の形状が加工容易な形状からより困難な形状になる順番で表示されてもよい。

20 【0053】上記の実施の形態例では、移動ブロック体が画面の最下位置に落下した時点で爆発し、或いは修復不可能なブロックを消去した時に爆発し、その爆発したブロックがプレーヤに衝突した時に、ミスと判定されてプレーヤの数が減少する。しかしながら、本発明は上記に限定されずに、例えば移動ブロック体が最下位置に落下した時にミスと判定されてプレーヤの数が減少するようにしてもよい。または、修復不可能なブロックを消去した時にもプレーヤの数が減少するようにしてもよい。

【0054】

30 【発明の効果】以上説明した通り、本発明によれば、単純な形状の落下ブロックに対して、与えられた問題ブロックの形状への加工を4方向から行うことを基本とする。問題ブロックの形状を幾何学的に認識し、最も効率的な落下ブロックの加工方法を検討しながらミサイルによる消去操作を行う必要があり、単純ではあるがバラエティに富んだゲームを提供することができる。

【0055】しかも、加工される落下ブロックのデータを常に更新して、ブロックが消去される度に問題ブロックとの照合を行うことで、少ない演算工程で問題ブロックがクリアされたか否かを検出することができる。

40 【0056】更に、問題ブロックが選択されてから、その4方向の照合用の問題ブロックのデータを予め演算により求めておくことで、落下ブロックの加工が始まってからリアルタイムに問題ブロックがクリアされたか否か及び修復不可能なブロックが消去されたか否かの演算を行うことが可能になる。

【0057】従って、低コストでそれほど高いCPUの能力を備えなくとも、ゲームプログラムの実行を可能にし、安価で興味深いゲーム装置を提供することができる。

【図面の簡単な説明】

50 【図1】本実施の形態例のゲーム装置の例の外観図である。

13

【図2】本実施の形態例のゲーム装置の内部構成の例を示す図である。

【図3】ゲームの進行に伴う画面の例を示す図である。

【図4】問題ブロックの他の例を示す図である。

【図5】問題ブロックデータと照合用に生成される問題ブロックデータとを示す図である。

【図6】メインメモリ内のデータ構成例を示す図である。

【図7】ゲーム全体のフローチャート図である。

【図8】イベントサブルーチンのフローチャート図である。

14

【図9】プレーヤサブルーチンのフローチャート図である。

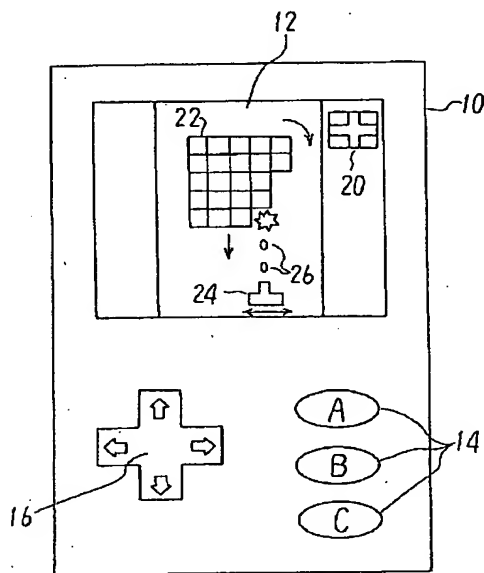
【図10】ミサイルサブルーチンのフローチャート図である。

【符号の説明】

- |       |                 |
|-------|-----------------|
| 10    | 筐体              |
| 12    | 液晶表示部           |
| 14、16 | 操作スイッチ          |
| 30    | CPU             |
| 32    | メインメモリ（記録媒体）    |
| 36    | ゲームカートリッジ（記録媒体） |

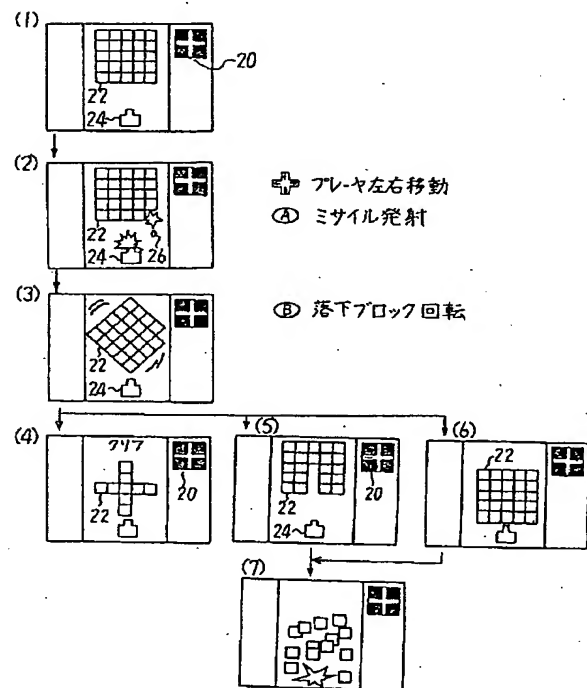
【図1】

ゲーム装置の外観図



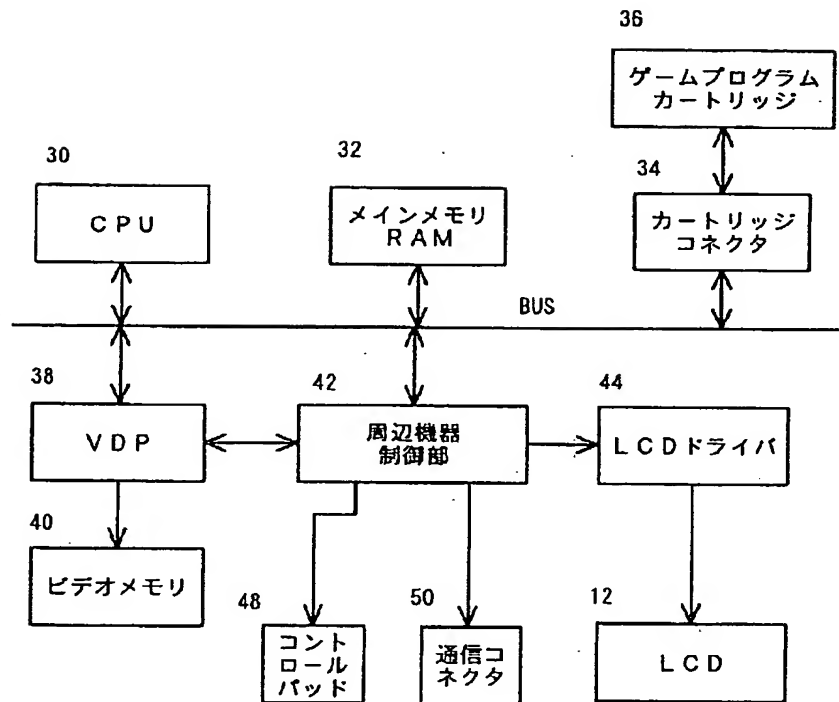
【図3】

ゲームの進行に伴う画面の例を示す図



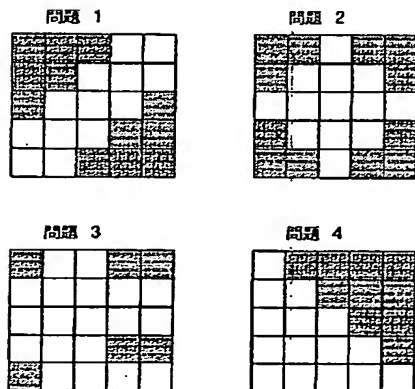
【図2】

ゲーム装置の構成図



【図4】

問題ブロックの例



【図5】

(1) 問題ブロックデータ

1	0	0	0	0
1	1	0	0	0
1	1	1	0	0
1	1	1	1	0
1	1	1	1	1

(2) 問題ブロックデータ0°

1	0	0	0	0
1	1	0	0	0
1	1	1	0	0
1	1	1	1	0
1	1	1	1	1

(3) 問題ブロックデータ90°

1	1	1	1	1
1	1	1	1	0
1	1	1	0	0
1	1	0	0	0
1	0	0	0	0

(4) 問題ブロックデータ180°

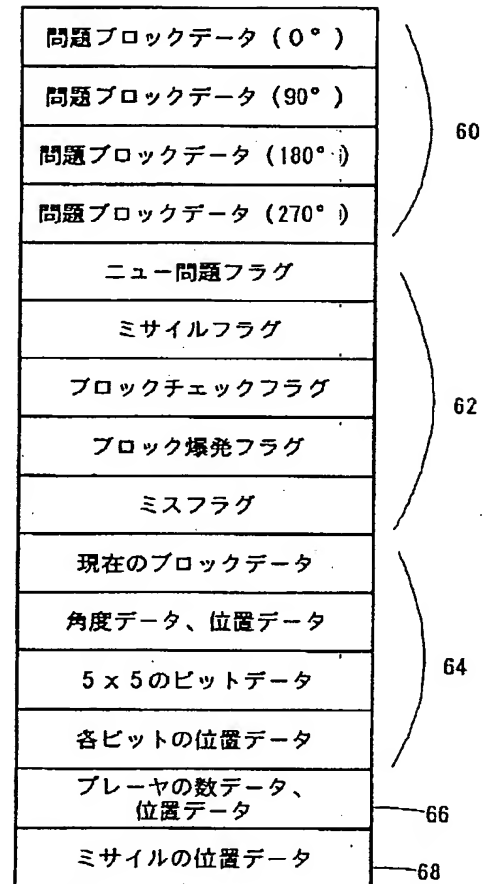
1	1	1	1	1
0	1	1	1	1
0	0	1	1	1
0	0	0	1	1
0	0	0	0	1

(5) 問題ブロックデータ270°

0	0	0	0	1
0	0	0	1	1
0	0	1	1	1
0	1	1	1	1
1	1	1	1	1

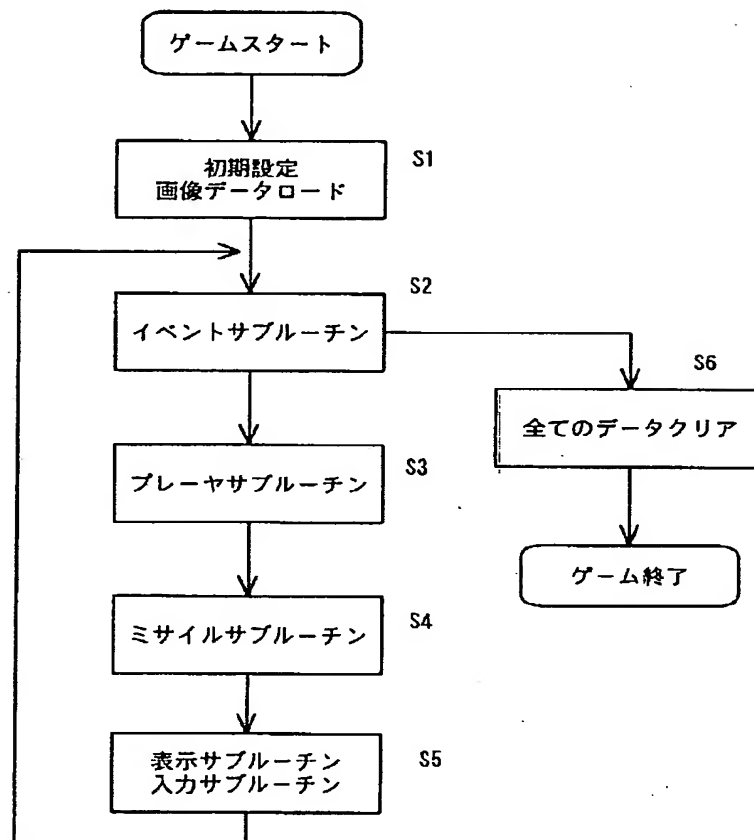
【図6】

## メインメモリ内のデータ構成

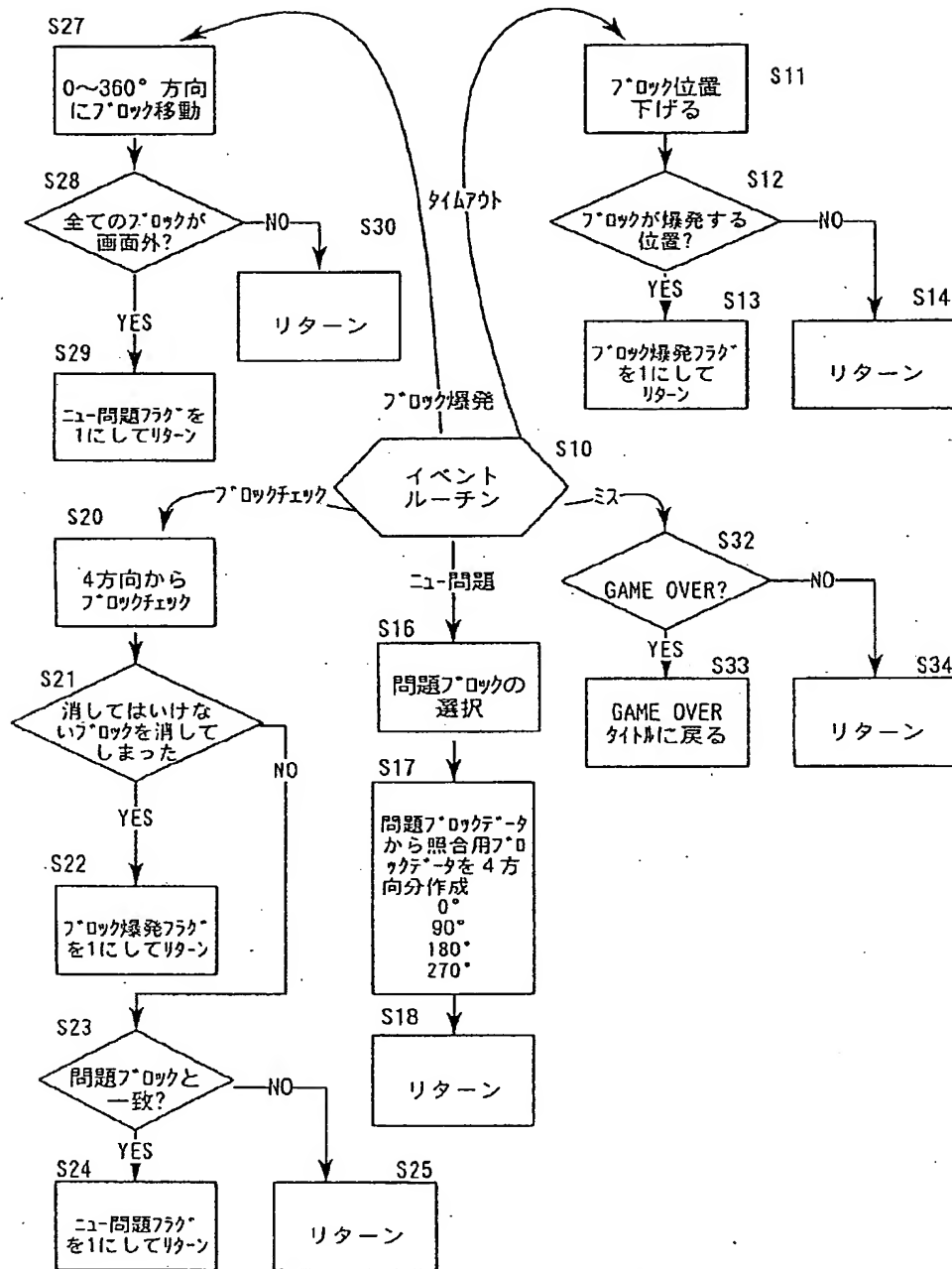


【図7】

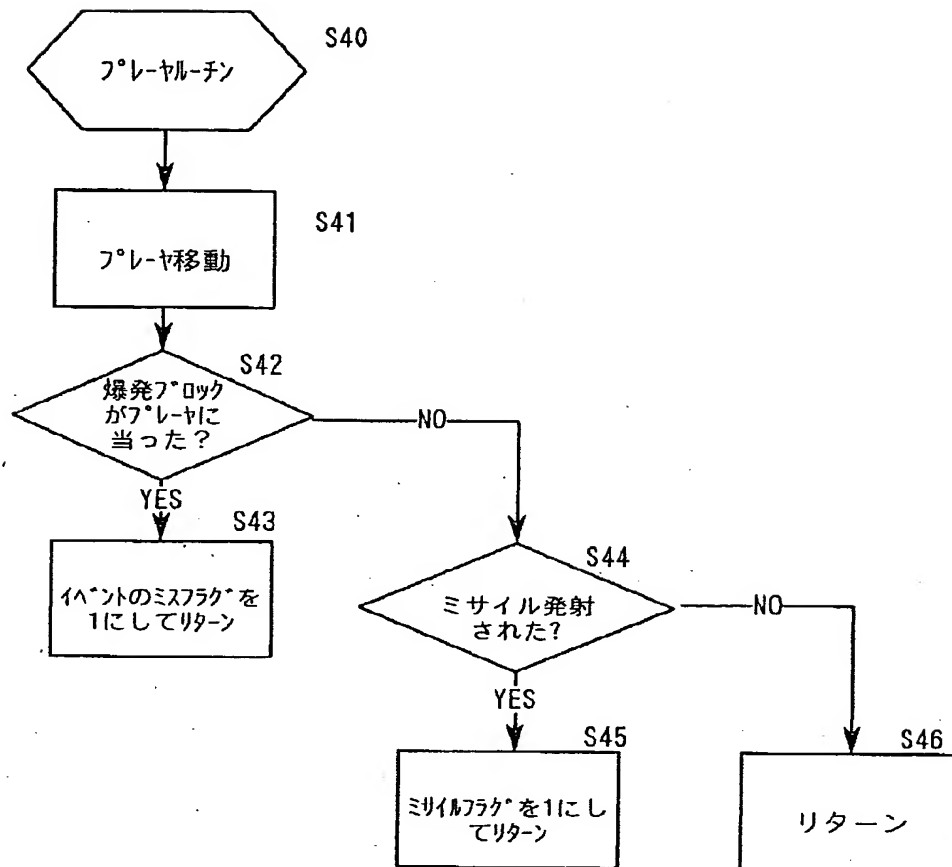
## ゲームプログラム全体のフローチャート



【図8】



【図9】



【図10】

